

# Coding for Cryptographic Security Enhancement using Stopping Sets

\*Willie K. Harrison, *Student Member, IEEE*, João Almeida, *Student Member, IEEE*, Steven W. McLaughlin, *Fellow, IEEE*, and João Barros, *Member, IEEE*

**Abstract**—In this paper we discuss the ability of channel codes to enhance cryptographic secrecy. Toward that end, we present the secrecy metric of degrees of freedom in an attacker's knowledge of the cryptogram, which is similar to equivocation. Using this notion of secrecy, we show how a specific practical channel coding system can be used to hide information about the ciphertext, thus increasing the difficulty of cryptographic attacks. The system setup is the wiretap channel model where transmitted data traverse through independent packet erasure channels with public feedback for authenticated ARQ (Automatic Repeat reQuest). The code design relies on puncturing nonsystematic low-density parity-check codes with the intent of inflicting an eavesdropper with stopping sets in the decoder. Furthermore, the design amplifies errors when stopping sets occur such that a receiver must guess all the channel-erased bits correctly to avoid an expected error rate of one half in the ciphertext. We extend previous results on the coding scheme by giving design criteria that reduces the effectiveness of a maximum-likelihood attack to that of a message-passing attack. We further extend security analysis to models with multiple receivers and collaborative attackers. Cryptographic security is enhanced in all these cases by exploiting properties of the physical-layer. The enhancement is accurately presented as a function of the degrees of freedom in the eavesdropper's knowledge of the ciphertext, and is even shown to be present when eavesdroppers have better channel quality than legitimate receivers.

## I. INTRODUCTION

### A. Cryptography and the Physical Layer

Many cryptosystems in place today measure security computationally. If all known attacks are computationally intractable, then the system is deemed to be secure. The chief failings of this notion of security are the assumptions placed on the attacker. First, it is assumed that the attacker has limited resources to confront the problem, even if those resources are state of the art. Second, it is assumed that the attacker uses attacks which are publicly known, even though a better attack may exist. Claude Shannon addressed these shortcomings by defining the notion of *perfect secrecy* [1]. If a secret message  $M$  is encrypted into a cryptogram  $E$  using a secret key  $K$ , then perfect secrecy is achieved if  $H(M|E) = H(M)$ . Shannon also proved that perfect secrecy is only attainable if the key is at least as long as  $M$ , which is clearly impractical. However, perfect secrecy also makes the limiting assumption that an attacker has access to an error-free cryptogram, which may not be the case in practice.

Aaron Wyner later introduced the wiretap channel model, along with a new condition for secrecy [2]. Let a message  $M$  of length  $k$  be encoded into a codeword  $X$  of length  $n$ , and then transmitted. The rate of the encoder is  $k/n$ . A legitimate

receiver obtains  $Y$  over the *main channel* denoted  $Q_m$ , and an eavesdropper obtains  $Z$  over a *wiretap channel* denoted  $Q_w$ . The secrecy condition is

$$\lim_{k \rightarrow \infty} \frac{I(M; Z)}{k} = 0. \quad (1)$$

Wyner showed that for rates up to the secrecy capacity  $C_s$ , encoders and decoders exist which can satisfy (1) and also achieve arbitrarily low probability of error for intended parties when  $X \rightarrow Y \rightarrow Z$  is a Markov chain. This is known as the *degraded* wiretap channel model. Csiszár and Körner [3] later generalized these results removing the degraded restriction, but still showing that  $C_s > 0$ , only if  $Q_m$  is *less noisy* than  $Q_w$ .

Understanding of the theoretically achievable secrecy rates of communication systems has continued to grow, as outlined in e.g. [4], [5], and [6]. But another of the main challenges in this area has been the design of practical systems which achieve the secrecy rates indicated by the theory. These systems exploit noise in the channel at the physical layer of the communications system. Practical designs maximizing the information-theoretic secrecy are not trivial. Most currently suffer from one or more of several drawbacks. For instance, code designs are oftentimes a function of specific channel parameters (channel state information or CSI) seen by legitimate receivers and eavesdroppers. Without accurate CSI, the results of these systems are not guaranteed; therefore, channels with varying or unknowable parameters present design issues. Other codes offer secrecy for only specific types of channels, or only when the eavesdropper's channel is degraded. Still other designs are impractical in the real world due to design complexity, necessary side information for legitimate decoding, or other limitations. Finally, the most glaring shortcoming of any scheme which derives security from the physical layer of a communications system, is that if an eavesdropper has a *better* channel than a legitimate receiver, the scheme is likely to fail. The extreme case is when an eavesdropper has a noise-free channel and  $Z = X$ . Clearly this necessitates any physical-layer security scheme to be coupled with some other protection in order to maintain secrecy in the worst case.

### B. Main Contributions

The intent of this paper is to develop the notion of *combined security* due to cryptography and channel coding, thus providing a more complete security solution. To accomplish this goal, we cast coding into a cryptographic enhancement role, and seek to prevent an attacker from obtaining a noise-free cryptogram using channel coding. We present a new

security metric for physical-layer schemes; namely, degrees of freedom  $D$  in an attacker's knowledge of the cryptogram. As a comparison, if bits in  $M$  are uniformly zero or one and independent and identically distributed (i.i.d.), then perfect secrecy implies  $D = k$ . In fact we show that  $H(X|Z) = E[D]$  for a specific case. Our notion of physical-layer security using  $D$  addresses the effectiveness of attacks on a cryptographic layer. To be more precise, our notion of security answers the practical question, how does the complexity of an attack on the cryptography change without perfect knowledge of the cryptogram?

It has been shown previously using correlation attacks on stream ciphers that certain cryptographic attacks are still possible even on noisy cryptograms, although a threshold on the noise level exists such that errors beyond the threshold cause the attack to fail [7], [8], [9], [10]. Practical schemes should provide enough confusion to exploit even the smallest amount of noise in an eavesdropper's received data to cause failure of these attacks on the cryptographic layer. Such systems should be robust to varying channel parameters, imperfect CSI at the encoder, and nondegraded system models. In fact, good designs still offer security enhancement to cryptography, even when attackers have an advantage in signal quality over legitimate receivers. Of course, all of this must be done while guaranteeing reliable communication between friendly parties.

Therefore, along with the new metric, this paper also analyzes combined cryptographic and physical-layer security in a *practical coding scheme* using degrees of freedom to characterize security. In [11], this scheme was shown to inflict a passive eavesdropper using a message-passing decoder with stopping sets with very high probability when a legitimate receiver and an eavesdropper view transmitted data through statistically independent packet erasure channels (PEC). The scheme relies on a nonsystematic low-density parity-check (LDPC) code design, with puncturing and interleaving steps in the encoder. Legitimate receivers are given access to an authenticated public feedback channel for Automatic Repeat-Request (ARQ). In this paper, we broaden the security analysis of the scheme given in [11] by addressing the following points.

- *Degrees of Freedom:* The system security is analyzed using the new metric. Computational secrecy is shown to grow exponentially with  $E[D]$ , which is also shown to be equal to  $H(X|Z)$  for the prescribed encoder.
- *Encoder Description:* End-to-end details of the encoder and decoder are provided, as well as simulation results which match theoretical expectations.
- *Optimization:* Design criteria are specified to maximize the degrees of freedom in the maximum-likelihood attack as well as the message-passing attack. This involves comparison of irregular LDPC codes with regular LDPC codes.
- *Extensions:* Security results are made general so as to apply to multiple receivers and multiple collaborative attackers. Ultimately, bounds on the increase in computational secrecy of an underlying cryptosystem are specified when the physical-layer encoding system is employed.

Ultimately, this scheme has very few design constraints, offers

enhanced cryptographic secrecy over a wide range of CSI parameters, and requires no secret key and no rate reduction in data transmission.

### C. Related Works

Our encoder makes use of fundamental practical design ideas which have been shown to offer secrecy. For example, our encoder employs nonsystematic LDPC codes in order to hide information bits and magnify coding errors. Secrecy properties of these codes have been studied in [12]. We further employ intentional puncturing of encoded bits, a technique shown to offer security in [13], [14]. Our scheme punctures with the goal of inducing *stopping sets* in an eavesdropper's received data. As a result, every transmitted bit is crucial for decoding. Our intent is to punish an eavesdropper for every missing piece of information. Finally, in order to distribute erasures throughout the data set, the encoder interleaves coded bits among several transmitted packets. Similar ideas of interleaving coded symbols have been used in [15], [16] in conjunction with wiretap codes developed in [17] to offer secrecy to various systems. The works [18], [19] give results for ARQ and feedback wiretap systems.

It can be argued that the first practical secrecy coding scheme was presented by Ozarow and Wyner in an extension of the original wiretap paper [4]. Here the general idea of partitioning a group code into cosets to achieve secrecy was first presented. This technique was shown to apply to LDPC codes much more recently in [17], and achieves the secrecy condition in (1) for noiseless main channels when the wiretap channel is either a binary erasure channel (BEC) or a binary symmetric channel (BSC). This work in LDPC codes for secrecy has been furthered in [20], where large-girth LDPC codes are considered, and shown to meet the secrecy constraint in (1) for noiseless main channel and BEC wiretap channel. A stronger notion of secrecy than (1) is also achieved for these codes in certain cases. Finally, it should be noted that Arkan's polar codes [21] can offer secrecy for general symmetric channels, although code construction is an issue for non-erasure channels. Schemes have been presented in [22] and [23] which achieve the secrecy capacity under the condition in (1), although these schemes only offer secrecy for degraded wiretap channels. Furthermore, design of these codes is heavily contingent on perfect CSI at the encoder.

Although our codes can be shown to achieve (1) only under certain puncturing criteria, the main contribution of the coding scheme presented here is the cryptographic security enhancements shown using degrees of freedom as a security metric. Our scheme is robust against imperfect CSI, and for that matter, undetected eavesdroppers. According to our knowledge, it is also the first *practical* secrecy scheme which can operate on the general wiretap channel (nondegraded case) when both  $Q_m$  and  $Q_w$  are erasure channels.

The rest of the paper is outlined as follows. In Section II, we discuss the system model for which our encoder is designed, which is an adaptation of the wiretap channel model from [2]. The precise definition of degrees of freedom is also given. Section III addresses background information regarding

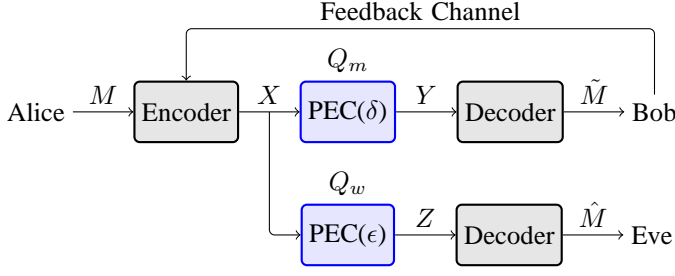


Fig. 1. Wiretap channel model with feedback assuming packet erasure channels for both the main channel  $Q_m$  and the wiretap channel  $Q_w$ .

LDPC codes and stopping sets. Our novel encoder and decoder designs are presented in Sections IV and V, respectively. Analysis of the security inherent in the system is then completed in Section VI for various scenarios, ultimately culminating in the most general case which encompasses multiple users and collaborating eavesdroppers. Finally, bounds regarding enhancements of cryptographic security are presented in Section VII along with end-to-end simulations of the system. Conclusions are provided in Section VIII.

## II. SYSTEM MODEL AND DEGREES OF FREEDOM

We begin by presenting the wiretap channel model [24] with the addition of feedback in Fig. 1. A user named Alice wishes to transmit an *encrypted* binary message  $M = (m^1, m^2, \dots, m^L)$  to a legitimate receiver named Bob, where  $m^i = (m_1^i, m_2^i, \dots, m_k^i) \in \mathcal{M}$  for  $i = 1, 2, \dots, L$ . It will be helpful to think of  $M$  as being broken up into  $L$  blocks of length  $k$ , where  $k$  is the dimension of the encoder to follow. The final block  $m^L$  can be filled by concatenating random bits if needed. Let us also define the blocklength  $n$  of the encoder. Then the coding rate is  $k/n$ . To be clear,  $n$  is the length of a codeword after it has been punctured. We will also assume that  $M$  has been compressed, so that all possible bit combinations are equally likely in the alphabet  $\mathcal{M}$ . Prior to transmission, Alice encodes  $M$ , resulting in a collection of  $\eta$  packets  $X = (x^1, x^2, \dots, x^\eta)$  for transmission. Bob receives the packets as  $Y$  through  $Q_m$ , a PEC with probability of erasure  $\delta$ . An eavesdropper named Eve obtains the packets  $Z$ , although through  $Q_w$ , an independent PEC with probability of erasure  $\epsilon$ . An obvious extension of this model is to consider correlated erasures in  $Q_m$  and  $Q_w$ ; however, in this paper we always assume erasures are statistically independent. Finally,  $\tilde{M}$  and  $\hat{M}$  are the respective estimates of  $M$  by Bob and Eve.

The encoder and decoder exploit the independent nature of erased packets across  $Q_m$  and  $Q_w$ . Of course, the system must guarantee that  $\tilde{M} = M$ , while at the same time making Eve as ignorant as possible. The authenticated feedback channel available to Bob plays a key role in accomplishing both of these endeavors. This public noiseless channel is used to request the retransmission of erased packets. Since it is authenticated, Alice is able to deduce whether Bob sent the request, and can detect any tampering with the data [25], which restricts Eve to passive status [26]. Requests by Bob are public, and there is no *secret key* employed at the physical layer. The

sole source of confusion for Eve is her own naturally occurring erasure pattern across  $Q_w$ .<sup>1</sup>

As mentioned in Section I, we define *physical-layer security* for this system with the cryptographic layer in mind. Cryptographic attacks often assume an attacker has the luxury of an error-free version of  $M$  (or even some of the plaintext), but our design aims to prevent this from occurring, by creating degrees of freedom in the attacker's knowledge of  $M$ .

**Definition 1.** The number of *degrees of freedom* in a received codeword is a random variable  $D$  which takes on the number of encoded symbols for which an eavesdropper has no information. Therefore, the probabilities of all symbol values on these  $D$  symbols are equally likely.

For binary codes with  $D = d$ , a codeword of length  $n$  can be any of  $2^d$  equally likely codewords, each mapping to a unique  $k$ -bit message in  $\mathcal{M}$ . Since we assume that the attacker knows the encoder, the maximum value of  $D$  is  $k$ , and can be shown to have an information-theoretic definition. Since an attacker has no knowledge of these bits, an average of  $2^{E[D]} - 1$  guesses must be made to obtain them. Using this reasoning, the goals of our physical-layer design are: first, to ensure that  $D = 0$  for Bob so that  $\tilde{M} = M$ ; second, to make  $D$  as large as possible for Eve; and third, to ensure that attacks on the cryptogram fail if  $\hat{M} \neq M$ .

## III. LDPC CODES AND STOPPING SETS

We employ LDPC codes [27] and exploit the phenomenon of stopping sets to obtain security from the physical layer. This section provides limited background of LDPC codes and stopping sets in order to establish the foundation upon which to present our encoder.

Let us define a general binary LDPC code  $C$  with block-length  $N$ , and dimension  $k$ . Note that this  $k$  is identical to  $k$  from section II, but  $N$  the blocklength of the LDPC code, is different from  $n$  the blocklength of the encoder because  $n$  is the codeword length after puncturing. The parity check matrix  $H$  fully defines the code, and is  $N - k \times N$ . We will find it helpful to think of  $H$  in terms of its corresponding Tanner graph  $G_C$  [28], [29]. The set of variable nodes is  $V = (v_1, v_2, \dots, v_N)$ , while the set of check nodes is  $U = (u_1, u_2, \dots, u_{N-k})$ . Variable nodes correspond to the  $N$  bits in a codeword. Checks correspond to rows in  $H$ , where the set of bits that participate in the check  $u_i$  is denoted  $\mathcal{N}_i = \{j : H_{i,j} = 1\}$  [28]. Then the  $i$ th check is calculated in GF(2) as  $u_i = \sum_{j \in \mathcal{N}_i} v_j = 0$ . The notation  $\mathcal{N}_{i,j}$  signifies all bits in the  $i$ th check except the  $j$ th bit. The  $j$ th variable node shares an edge with the  $i$ th check node in  $G_C$  if and only if  $j \in \mathcal{N}_i$ . The Tanner graph for a simple example is shown in Fig. 2.

Decoding of an LDPC codeword over a BEC can be accomplished using maximum-likelihood (ML) decoding [30], by solving a system of equations. However, the iterative message-passing (MP) decoder is commonly used due to its computational efficiency. We briefly explain both decoders.

<sup>1</sup>It is noted that results in Section VI are provided for this system, as well as the more general model which allows an arbitrary number of legitimate receivers and eavesdroppers.

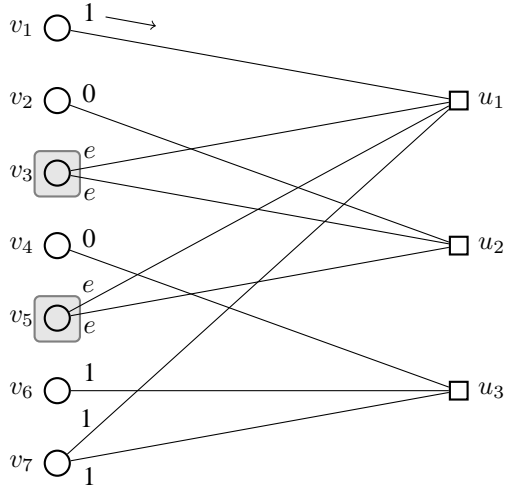


Fig. 2. Tanner graph for MP decoding over the BEC with a highlighted stopping set due to erasures at variable nodes  $v_3$  and  $v_5$ .

### A. Maximum-Likelihood Decoding

Let us consider an LDPC codeword  $x \in C$  transmitted over a BEC and let  $y$  denote the received codeword. Note that  $x_i \in \{0, 1\}$  and  $y_i \in \{0, 1, e\}$  where  $e$  signifies an erased bit. We let  $\mathcal{K}$  denote the set of known bits in  $y$ , and  $\bar{\mathcal{K}}$  denote the set of erased bits in  $y$ . Furthermore,  $H_{\mathcal{K}}$  and  $H_{\bar{\mathcal{K}}}$  can be understood to be matrices formed by the columns of  $H$  indexed by  $\mathcal{K}$  and  $\bar{\mathcal{K}}$ , respectively. Similarly,  $x_{\mathcal{K}}$  and  $x_{\bar{\mathcal{K}}}$  are vectors composed of only the bits indexed by the respective sets  $\mathcal{K}$  and  $\bar{\mathcal{K}}$ .

Clearly,  $0 = Hx^T = H_{\mathcal{K}}x_{\mathcal{K}}^T + H_{\bar{\mathcal{K}}}x_{\bar{\mathcal{K}}}^T$ , where  $x_{\mathcal{K}} = y_{\mathcal{K}}$ , and thus  $H_{\mathcal{K}}x_{\mathcal{K}}^T = z^T$  is known. The maximum likelihood decoder must then solve for the channel-erased bits  $x_{\bar{\mathcal{K}}}$  using the system of equations given by

$$H_{\bar{\mathcal{K}}}x_{\bar{\mathcal{K}}}^T = z^T. \quad (2)$$

This system has a unique solution when the erased bits are such that the columns of  $H_{\bar{\mathcal{K}}}$  are linearly independent [31]. We can obtain a bound from this statement which we will use to analyze security in the worst-case.

**Proposition 1.** *For a linear code  $C$  with blocklength  $N$  and dimension  $k$ , the ML decoder over the BEC cannot have a unique solution if the number of erasures exceeds  $N - k$ , that is if  $|\bar{\mathcal{K}}| > N - k$ .*

*Proof:* The rank of  $H_{\bar{\mathcal{K}}}$  equals the number of linearly independent rows or columns of the matrix ([32], pg. 244). Since  $N - k$  is the number of rows in  $H$ , the rank of  $H_{\bar{\mathcal{K}}}$  can never exceed  $N - k$ , and thus the ML decoder cannot produce a unique solution when  $|\bar{\mathcal{K}}| > N - k$ . ■

In fact, when the number of erasures exceeds  $N - k$ , the system in (2) will be such that the degrees of freedom in the ML decoder  $D_{ML} \geq |\bar{\mathcal{K}}| - (N - k)$ , where we achieve equality if there are  $N - k$  linearly independent columns in  $H_{\bar{\mathcal{K}}}$  [30]. In any case,  $D_{ML}$  is equal to the difference in the number of erased bits, and the number of linearly independent columns of  $H_{\bar{\mathcal{K}}}$ , and is zero if this difference is negative. This definition

clearly satisfies the notion of degrees of freedom from Definition 1 for this decoder. Thus we see that the effectiveness of the decoder is strictly bounded by the redundancy of the code. While faster methods have been discovered for solving a linear system of equations, the straightforward decoder is known to have complexity  $((1 - R)\beta + \gamma\delta)\delta^2 N^3$ , where  $R$  is the rate of the code,  $\beta$  and  $\gamma$  are constants which are also a function of the elimination algorithm chosen to solve the system of equations,  $\delta$  is the erasure probability in the channel, and  $N$  is the blocklength of the code [30].

### B. Message-Passing Decoding

Let  $C$ ,  $x$ , and  $y$  hold the same definitions as for the ML decoder. The MP decoder is an iterative decoder based on the Tanner graph representation of  $C$ . The decoding process passes *messages* between  $U$  and  $V$  along the edges of  $G_C$ . One version of the decoder is given as Algorithm 1 (adapted from [31]). The number of degrees of freedom in the MP decoder  $D_{MP}$  is the cardinality of the smallest set of bit values that must be supplied in order to decode all remaining bits. If the decoder succeeds, then  $D_{MP} = 0$ . Clearly, this maintains the definition of degrees of freedom given in Definition 1 when restricted to this decoder, because any bit combination of these  $D_{MP}$  values decodes to a valid codeword, and each is equally likely without further information. A bound on the correction capabilities of the MP decoder is given by the following proposition.

**Proposition 2.** *The MP decoder over the BEC can correct no more than  $N - k$  erasures.*

*Proof:* In Algorithm 1, each check node can correct at most one variable node, and  $|U| = N - k$ . ■

The MP decoder is suboptimal compared with the ML decoder, although the MP decoder has linear complexity in the blocklength [28]. A more detailed comparison of the two decoders is offered in [33].

---

#### Algorithm 1 Message-Passing Decoder over the BEC [31].

---

- 1: **Initialize:** For  $y_i \neq e$ , set  $v_i = y_i$  and declare all such variable nodes as known.
  - 2: **if** (No variable nodes are known and no check node has degree one) **then**
  - 3:     Output the (possibly partial) codeword and stop.
  - 4: **else**
  - 5:     Delete all known variable nodes along with their adjacent edges.
  - 6: **end if**
  - 7: For each variable node  $v_j$  connected to a degree one check node  $u_i$ , declare  $v_j$  as known and set  $v_j = \sum_{k \in \mathcal{N}_{i,j}} v_k$ . Jump to 2.
- 

### C. Stopping Sets

In order to make  $D$  as large as possible for our system when an eavesdropper uses an MP decoder, we would like to design the encoder block from Fig. 1 so that every bit erased by the

channel adds a degree of freedom to the decoder. Stopping sets provide a means of accomplishing this task.

**Definition 2** (Di, et. al. [34]). A *stopping set* is a set  $S \subseteq V$  such that all check nodes in  $N(S)$  are connected to  $S$  by at least two edges, where  $N(S)$  signifies the *neighborhood* of  $S$  and is defined as the set of all adjacent nodes to any member of  $S$  in  $G_C$ .

Notice that the empty set, by definition, is a stopping set, as is any union of stopping sets. Thus, any set of variable nodes has a unique maximal stopping set in it.<sup>2</sup> See Fig. 2 for a simple example; clearly the erasures cannot be resolved using Algorithm 1. This gives way to the following lemma, proved in [34].

**Lemma 1** (Di et. al. [34], Lemma 1.1). *Let  $G$  be the Tanner graph defined by the parity check matrix  $H$  of a binary linear block code  $C$ , and assume that  $C$  is used to transmit over the BEC. Let  $A$  be the set of erased bits in the received codeword. Then, using Algorithm 1 on  $G$ , the set of erasures which remain after decoding comprise the unique maximal stopping set in  $A$ .*

Since stopping sets cause the MP decoder to fail, puncturing in the encoder will be done with an attempt to inflict Eve with stopping sets. However, the ML decoder will still succeed, even in the presence of stopping sets, as long as the erased bits have linearly independent columns in  $H$ . We account for both decoders in our design by using a particular ensemble of LDPC codes where  $D_{MP}$  can be made equal to  $D_{ML}$ , thus ensuring secrecy regardless of the decoder used by Eve. The simplicity of MP decoding is also preserved for all legitimate receivers.<sup>3</sup>

#### IV. ENCODER

The encoder design is based on the fact that  $I(M; Z) \leq I(M; X)$  because processing cannot increase information, and  $M \rightarrow X \rightarrow Z$  is a Markov process [37]. The key idea in the decoder is to reduce  $X$  to the decoding threshold. In other words,  $X$  can be used to recover  $M$  by design, but if any erasures remain in  $Z$  following transmission, unique decodability is not possible. Proper design maximizes  $D$  for Eve. The stages of encoding are portrayed in Fig. 3, where each stage fulfills a specific purpose within the overall goals of obtaining secrecy and reliability. The following principles are addressed in the design of this encoder.

- Bits of  $M$  are hidden from immediate access in the decoded words using nonsystematic LDPC codes.
- Scrambling prior to coding magnifies errors due to the physical layer of the communication system.
- The error-correction capabilities of the LDPC code are restricted by intentional puncturing of encoded bits. (Bob obtains reliability through ARQ, rather than error correction.)

<sup>2</sup>For our purposes, we will sometimes ignore the empty set as a stopping set and say that a set  $A$  contains no stopping sets, meaning that the maximal stopping set in  $A$  is  $\emptyset$ .

<sup>3</sup>For further information on stopping sets as they relate to LDPC code ensembles, see [35] and [36].

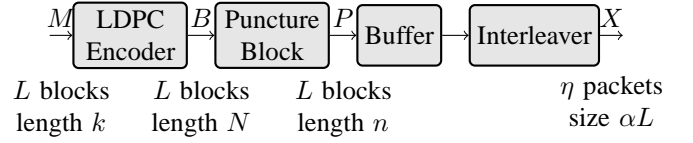


Fig. 3. Detailed block diagram of the encoder. Number and size of blocks or packets are indicated at each step.

- Bits from encoded blocks are interleaved amongst several transmitted packets so that a single erased packet results in erasures in many encoded blocks of data.

#### A. Nonsystematic LDPC Codes

Recall from Section II that  $M = (m^1, m^2, \dots, m^L)$ , where  $m^i = (m_1^i, m_2^i, \dots, m_k^i) \in \mathcal{M}$  for  $i = 1, 2, \dots, L$ . These  $L$  blocks of encrypted message form the input to the nonsystematic LDPC encoder with blocklength  $N$  and dimension  $k$ . The output of the LDPC encoder is given as  $L$  codewords of length  $N$ , denoted as  $B = (b^1, b^2, \dots, b^L)$  where each vector  $b^i = (b_1^i, b_2^i, \dots, b_N^i)$ . Certainly, if the code  $C$  were systematic, then the bits of  $m^i$  would appear explicitly in the encoded block  $b^i$ . For secrecy purposes, nonsystematic codes are employed.

Nonsystematic LDPC coding is typically implemented as a two stage process to improve encoder complexity [38], [39], [12]. Let  $S$  be an invertible  $k \times k$  scrambling matrix in  $\text{GF}(2)$ , and let  $G$  be a  $k \times N$  systematic generator matrix. Let  $m$  be a length- $k$  message. Then our LDPC encoding process applies the scrambling matrix to  $m$  as

$$m' = mS. \quad (3)$$

The data are then encoded using  $G$  by  $b = m'G$  to obtain a length- $N$  block of encoded data. Clearly at the decoder the inverse operation first requires the bits of  $b$  to be obtained through either MP or ML decoding. Since  $G$  is systematic, the bits of  $m'$  are explicit in  $b$ . The bits of  $m$  can then be found by applying the inverse of  $S$  in the descrambling operation

$$m = m'S^{-1}. \quad (4)$$

This process amplifies errors in the decoding process as a function of the sparsity of  $S^{-1}$ . Note that  $S^{-1}$  can be obtained through e.g. LU decomposition [32], with modifications for  $\text{GF}(2)$ . In our experience, randomly generated scrambling matrices which are nonsingular are likely to have inverses with just less than 50% of the entries equal to one on average. If  $S$  matrices are randomly generated until one can be inverted to obtain  $S^{-1}$ , the resulting despreading operation is enough to cause even a single error in  $m'$  to result in roughly a 50% error rate in  $m$  as shown in Section VII. Although this can be made more precise, the result is intuitive because a bit in  $m$  is a linear combination of bits in  $m'$ . Thus, if there are an odd number of bits in error in a given combination of say  $m_i$ , then that bit will be in error. On average, the row weight in  $S^{-1}$  is approximately  $k/2$ , and the expectation of  $k/2$  bits in error holds for any number of errors in  $m'$ .

Since only one  $(S, S^{-1})$  pair need be used by the system, the matrices can be generated off-line, which does not affect encoding and decoding complexity. However, the complexity of both the encoder and the decoder is increased due to the matrix multiplications in (3) and (4). Both of these operations are  $\mathcal{O}(k^3)$ . General systematic encoder complexity is  $\mathcal{O}(N^2)$  because  $G$  is not sparse by design [28], although improvements can be made using appropriate preprocessing as outlined in [31]. The encoding technique specified in [31] gives encoder complexity of  $\mathcal{O}(N+g^2)$  where  $g$  is the *gap* in an approximate lower triangular form of the parity check matrix and is less than  $N - k$ . The complexities for the ML and MP decoders are given in Sections III-A and III-B as  $\mathcal{O}(N^3)$  and  $\mathcal{O}(N)$ , respectively.

### B. Puncturing

The next step in the encoding process is to apply a puncturing pattern to each codeword in  $B$ . Let the puncturing pattern  $R \in V$  indicate which bits in each  $b^i$  are to be punctured. Recall that  $V$  is the set of variable nodes in the Tanner graph  $G_C$ . The punctured blocks  $P = (p^1, p^2, \dots, p^L)$ , where each  $p^i = (p_1^i, p_2^i, \dots, p_n^i)$  are shown in Fig. 3 to have length  $n$ , which was defined in Section II to be the blocklength of the encoder. All bits which are not punctured belong to the set  $Q$  so that  $V = R + Q$ ; therefore, the length of each block in  $P$  is equal to  $|Q| = n$ . The puncturing pattern is chosen in order to induce stopping sets in an eavesdropper's received data.

**Definition 3.** A puncturing pattern  $R$  is deemed *acceptable* if and only if there are no stopping sets in  $R$ , and  $R + v$  contains some nonempty stopping set  $S_v$  for every variable node  $v \in Q$ .

Such a set  $R$  can be constructed using the random technique outlined in Algorithm 2, which also calls Algorithm 3 in order to check for stopping sets in a computationally tractable manner [11].

---

**Algorithm 2** Finds an acceptable puncturing pattern  $R$  within the set of all variable nodes  $V$ .

---

```

1: Initialize:  $R = v$ , for a randomly chosen  $v \in V$ , and  $Q = \emptyset$ .
2: if  $(V \setminus (R \cup Q) \neq \emptyset)$  then
3:   Choose another  $v$  randomly from  $V \setminus (R \cup Q)$ .
4:   Run Algorithm 3 with  $A = R + v$  to check for stopping sets.
5:   if  $(R + v$  has a stopping set, i.e. Algorithm 3 returns true) then
6:      $Q = Q + v$ .
7:   else
8:      $R = R + v$ .
9:   end if
10:  Jump to 2.
11: else
12:  Terminate.
13: end if
```

---



---

**Algorithm 3** Checks for the existence of stopping sets in a subset of variable nodes,  $A \subseteq V$  [11].

---

```

1: Initialize:  $S = A$ 
2: if  $(S \neq \emptyset)$  then
3:   Induce subgraph  $G'$  in  $G$  using  $(S \cup N(S))$ .
4:   if  $(\exists$  a check node in  $G'$  with degree 1) then
5:     Delete variable nodes from  $S$  which are adjacent to check nodes of degree 1 in  $G'$ , jump to 2.
6:   else
7:     Return true.  $S$  is the maximal nonempty stopping set in  $A$ .
8:   end if
9: else
10:  Return false. There is no nonempty stopping set in  $A$ .
11: end if
```

---

**Lemma 2.** The output of Algorithm 2 is always an acceptable puncturing pattern  $R$  as defined in Definition 3.

*Proof:* We must first show that upon completion of Algorithm 2, there are no stopping sets in  $R$ . Assume for a contradiction that  $R$  has a stopping set. Then there is a bit  $v \in R$  which when added to  $R$  during the construction process, caused a stopping set to first appear. Then by Algorithm 2,  $v \notin R$ . This provides the contradiction. It remains to be proved that Algorithm 3 operates as expected.

**Proposition 3.** Algorithm 3 always returns true when  $A$  has a nonempty stopping set, and always returns false otherwise.

*Proof of Proposition:* Suppose that the bits in  $A$  were actually erasures over the BEC, and Algorithm 1 was used to decode. Realize that erasures recovered in the  $i$ th iteration of Algorithm 1 correspond exactly to the nodes deleted in the  $i$ th iteration of Algorithm 3. If all bits can be resolved using MP decoding then all nodes will be deleted in Algorithm 3, and false is returned. If, however, MP decoding returns a partial codeword, then Algorithm 3 will return true because all remaining bits have degree greater than one in the induced subgraph  $G'$ . Therefore, by Lemma 1, the remaining nodes comprise the maximal stopping set of  $A$ . ■

To complete the proof of Lemma 2, we must also show that for any  $v \in Q$ ,  $R + v$  has a nonempty stopping set. Since in Algorithm 2 every  $v \in Q$  is such that for some subset  $R' \subseteq R$ ,  $R' + v$  has a stopping set, therefore  $R + v$  has a stopping set for any  $v \in Q$ . ■

Thus, puncturing according to  $R$  in each  $b^i$  for  $i = 1, 2, \dots, L$ , guarantees that every bit in each  $p^i$  is crucial for successful MP decoding.

Complexity of Algorithm 2 is linear in the blocklength  $N$ , because it chooses  $N - 1$  bits in a random order, and calls Algorithm 3 after each choice. The complexity of Algorithm 3 in the worst case, is quadratic in  $|U| = N - k$  the number of check nodes in  $G_C$ . Line 5 of the algorithm will be repeated a maximum of  $\sum_{i=1}^{|U|} i = \frac{|U|^2 + |U|}{2}$  times if a single node is deleted each time the line is executed. Therefore, the complexity of finding an acceptable puncturing pattern  $R$  is

at most quadratic in  $|U|$ , and linear in  $N$ , i.e. has complexity  $\mathcal{O}(N|U|^2)$ . Thus the algorithm can be used in practical system design to compute  $R$  off-line.

### C. Regular vs. Irregular Codes

The overall rate  $k/n$  of the nonsystematic and punctured code is a function of the rate of the systematic LDPC code, and  $|R|$ . Simulations have shown that the size of  $R$  is very much a function of the degree distribution on  $C$ , although the exact relationship is still unknown.

**Example 1.** Let  $C$  be a regular rate-1/2 code with  $N = 1000$ ,  $w_c = 4$ , and  $w_r = 8$ , where  $w_c$  and  $w_r$  are the fixed column and row weights of the parity check matrix, respectively. The size of  $|R|$  appears to be Gaussian-distributed for this family of codes with a mean size of approximately 436, with variance roughly equal to 15. Let us examine, however, an irregular ensemble with the same rate and blocklength, but having the following edge degree distribution pair:  $\eta(x) = 0.32660x + 0.11960x^2 + 0.18393x^3 + 0.36988x^4$  on variable node weights, and  $\chi(x) = 0.78555x^5 + 0.21445x^6$  on check node weights (see [28] pg. 664), where  $H$  is formed using the socket approach given in [30]. Here the distribution on  $|R|$  is much tighter, ranging from 496 to 500. The size on  $R$  is equal to 500 with probability roughly equal to 0.1, 499 with probability around 0.56, and 498 with probability near 0.26. Thus with some degree of confidence, we can claim that for this rate-1/2 irregular code ensemble the random technique given in Algorithm 2 yields a puncturing pattern with size nearly equal (and equal in some cases) to  $N - k$ .

As a direct result, a puncturing pattern generated for the irregular code of the example has a unique property. Namely, that for some patterns  $D_{MP} = D_{ML}$ .

**Lemma 3.** Let  $R_c$  denote the indices of the channel-erased bits of  $p^i$ , and  $D_{MP}$  and  $D_{ML}$  denote the degrees of freedom using MP decoding and ML decoding, respectively. If an irregular LDPC code is employed over the BEC with intentional puncturing determined by Algorithm 2 in which  $|R| = N - k$ , then  $D_{ML} = D_{MP} = |R_c|$ .

*Proof:* The ML portion of this lemma follows from Proposition 1, i.e. that the system of equations in (2) can resolve a maximum of  $N - k$  erasures. Since  $|R| = N - k$ , any erasure by the channel is guaranteed to give a degree of freedom in the decoder. The MP case is the same because by Proposition 2, the MP decoder can correct at most  $N - k$  erasures. Thus any bits erased by the channel (or perhaps another set of bits of equal size) must be guessed in order to decode. Therefore, the effectiveness of the ML decoder is equal to that of the MP decoder when  $|R| = N - k$ . ■

It should be noted that if the sum of systematic bits in  $R + R_c$  is less than  $D$ , a brute-force attack on these bits might be more appealing to an attacker than decoding the entire codeword. To cover this possibility,  $D$  can be thought of as the minimum between the number of systematic bits missing to the eavesdropper, and the degrees of freedom in the decoder. Although, in practice the number of systematic bits

removed through puncturing or erased by the channel usually exceeds the degrees of freedom in the decoder.

### D. Interleaving

The role of the interleaver is to ensure that all packets must be obtained error-free for successful decoding in any and all encoded blocks. To do this, we construct a collection of  $\eta$  packets to be transmitted  $X = (x^1, x^2, \dots, x^\eta)$  in the following manner. Alice defines  $\alpha$  a small positive integer which is assumed to divide  $n$  (not necessary but convenient for notation and analysis) such that  $\eta = n/\alpha$ , and the  $i$ th packet is formed as

$$\begin{aligned} x^i &= (x_1^i, x_2^i, \dots, x_{\alpha L}^i) \\ &= (p_{(i-1)\alpha+1}^1, \dots, p_{(i-1)\alpha+1}^1, p_{(i-1)\alpha+1}^2, \dots, p_{i\alpha}^2, \dots, \\ &\quad p_{(i-1)\alpha+1}^L, \dots, p_{i\alpha}^L). \end{aligned} \quad (5)$$

for  $i = 1, 2, \dots, \eta$ . In words, we form the packet  $x^i$  by concatenating  $\alpha$  bits from each encoded and punctured block  $p^j$  for  $j = 1, 2, \dots, L$ . Therefore, a single erased packet causes  $\alpha$  erasures in each punctured block at the decoder. Since we have designed  $R$  so that any erasure of a bit in  $p^j$  results in MP decoding failure, we can be assured that any erased packet will cause all  $L$  blocks to fail in the MP decoder due to this interleaving. If  $R$  can be designed so that  $|R| = N - k$ , then the same result holds for ML decoding by Lemma 3.

**Corollary 1.** If  $|R| = N - k$  and packets are formed according to (5), then the number of degrees of freedom in the  $i$ th codeword is  $D_{ML}^i = D_{MP}^i = |R_c^i| = \alpha|R_p|$  for  $i = 1, 2, \dots, L$ , where  $R_p$  is a list of all erased packets. Furthermore,  $D_{ML}^i = D_{MP}^j \forall i, j$ .

*Proof:* The first part is trivial and follows directly from Lemma 3 and (5). We see that  $D_{ML}^i = D_{MP}^j$  because a missing packet means exactly  $\alpha$  degrees of freedom in each block, irrespective of decoder choice. ■

## V. DECODER FOR LEGITIMATE USERS

The decoder for legitimate users is simply the inverse of all encoder operations. A user can decode all data as long as every packet is received error-free. Legitimate users make use of the authenticated feedback channel to request retransmission of packets erased in the main channel during transmission. Time delay and queueing aspects of ARQ protocols are well-addressed in the literature, e.g. [40] and its references. The decoding process is shown pictorially in Fig. 4. Once all packets are obtained in  $Y$ , the bits are deinterleaved back into their intentionally punctured codewords  $\tilde{P}$ . The MP decoder is then guaranteed to decode the puncturing in linear time with the blocklength to obtain  $\tilde{B}$  [28], and the inverse of the scrambling matrix is applied to the systematic decoded bits using (4) to obtain  $\tilde{M}$ . Once all packets are known, this decoder guarantees that  $\tilde{M} = M$ .

## VI. SECURITY AGAINST WIRETAPPERS

An eavesdropper can decode the data using Bob's decoder in Fig. 4 if all packets are obtained error-free. The independence

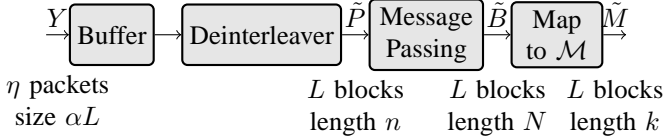


Fig. 4. Detailed block diagram of Bob's decoder. Number and size of blocks or packets are indicated at each step.

of  $Q_m$  and  $Q_w$ , however, prevents Eve from receiving packets as a function of  $\delta$  and  $\epsilon$ , the respective probabilities of erasures in  $Q_m$  and  $Q_w$ . Let  $R_{ef}$  be the event that a single packet is *received error-free* by at least one eavesdropper after all retransmissions of the packet requested by any legitimate receiver have been filled. This section shows the blanket security effect of our encoder over nearly the entire region of possible  $(\delta, \epsilon)$  pairs by completely characterizing  $D$  for the system. We first show  $D$  to be binomially distributed, and then provide security results for all scenarios studied as a function of  $R_{ef}$ . Expressions for  $R_{ef}$  follow for the wiretap channel case, the broadcast scenario with  $m$  intended receivers, the case with  $l$  collaborating eavesdroppers, and the most general case with both  $m$  legitimate receivers and  $l$  collaborating eavesdroppers. For cases beyond the simple wiretap scenario, all  $m$  legitimate receivers are given access to the feedback channel, and all  $l$  eavesdroppers are restricted to passive status through authentication on the channel. Retransmissions in the ARQ protocol are executed only after requests are received from all legitimate parties.

Since proper design of the encoder was shown to cause  $D$  to have the same realization for every codeword and be independent of the decoder in Corollary 1, we understand  $D$  to represent the degrees of freedom in every codeword assuming either the ML or MP decoder for the rest of the paper.

#### A. General Security Theorems

**Lemma 4.** *The random variable  $D$  which governs the number of degrees of freedom in a received codeword is a scaled binomial random variable. Thus, for  $1 \leq \beta \leq \alpha\eta$ ,*

$$\Pr(D \geq \beta) = 1 - \sum_{i=0}^{\lceil \beta/\alpha \rceil - 1} \binom{\eta}{i} (1 - \Pr(R_{ef}))^i \Pr(R_{ef})^{\eta-i}. \quad (6)$$

*Proof:* By definition, packets are erased for eavesdroppers with probability  $(1 - \Pr(R_{ef}))$ . Since there are  $\eta$  independent Bernoulli trials, each identically distributed, the sum of erased packets  $|R_p|$  is a binomial random variable with parameters  $\eta$  and  $(1 - \Pr(R_{ef}))$  [41]. Then, by Corollary 3,  $D = \alpha|R_p|$  where  $\alpha$  bits from every codeword are sorted into each packet. Thus,  $D$  is a scaled binomial random variable; specifically  $D \sim \text{Bin}(\eta, 1 - \Pr(R_{ef}))\alpha$ . Since  $D = \alpha|R_p|$ , then  $D \geq \beta$  implies that  $\alpha|R_p| \geq \beta$ . Clearly, this requires that  $|R_p| \geq \lceil \beta/\alpha \rceil$ . The result in (6) follows directly. ■

The expected value is therefore known due to the binomial structure of  $D$ . We also prove an important property in regards to  $E[D]$ .

**Theorem 1.** *If  $|R| = N - k$  in the encoder, then  $k/n = 1$ , and  $E[D] = H(X|Z) = (1 - \Pr(R_{ef}))n$ .*

*Proof:* Since  $|R| = N - k$ , then  $n = |Q| = N - |R| = k$ . Let us consider the model for a single codeword ( $L = 1$ ). We can then assume  $\eta$  independent uses of a PEC with packets of length  $\alpha$ . Let  $X$  be the input to the channel, and  $Z$  the output, where  $\alpha$  bits are erased with probability  $(1 - \Pr(R_{ef}))$  or received error-free with probability  $\Pr(R_{ef})$  with each channel use. The input distribution on  $\alpha$  bits is uniform because the input distribution on  $M$  is uniform, and the encoding function of rate one forms a bijection on  $k$  bits. Thus,  $H(X) = \alpha$ . Clearly  $H(Z|X) = H(1 - \Pr(R_{ef}))$ , and  $H(Z) = H(1 - \Pr(R_{ef})) + \Pr(R_{ef})\alpha$  (see [37], pg. 188). Then,

$$H(X|Z) = H(Z|X) - H(Z) + H(X) \quad (7)$$

$$= \alpha(1 - \Pr(R_{ef})). \quad (8)$$

Therefore, with  $\eta$  independent uses of the channel (one for each packet),  $H(X|Z) = (1 - \Pr(R_{ef}))\eta\alpha = (1 - \Pr(R_{ef}))n$ . Since the mean of a binomial random variable is the product of its two parameters,  $E[D/\alpha] = (1 - \Pr(R_{ef}))\eta$ , and therefore

$$E[D] = (1 - \Pr(R_{ef}))\eta\alpha = (1 - \Pr(R_{ef}))n. \quad (9)$$

Thus we see that  $E[D]$  is equal to the information-theoretic value of equivocation when the puncturing is accomplished so that  $|R| = N - k$ . Therefore, *perfect* secrecy is obtained when  $E[D] = k$ . Of course, this occurs when  $\Pr(R_{ef}) = 0$ , which implies that the eavesdropper obtains zero packets. Thus, this scheme cannot achieve perfect secrecy. However, it can be shown using the achievable rates in [4] that  $E[D]$  approaches the maximum achievable equivocation for  $k/n = 1$ . These results now require expressions for  $\Pr(R_{ef})$  to complete the security characterization in  $D$ . ■

#### B. One Receiver and One Wiretapper

The simplest case matches the setup given in Fig. 1, and was originally proved in [11].

**Lemma 5** (Harrison, et. al. [11]). *In the wiretap channel scenario with feedback, the probability that Eve obtains a single transmitted packet is given as*

$$\Pr(R_{ef}) = \frac{1 - \epsilon}{1 - \epsilon\delta}. \quad (10)$$

Intuition of security for the wiretap channel in terms of  $D$  can be gained by using the expression for  $\Pr(R_{ef})$  in (10) to plot (6) for different values of  $\beta$ ,  $\alpha$ , and  $\eta$ . Fig. 5 shows  $\Pr(D \geq 1)$  for  $\eta = 100$ . Note that when  $\beta = 1$ ,  $\alpha$  is not required to evaluate (6). This case is provided to show the plateau and falloff regions in the  $(\delta, \epsilon)$  grid for  $\Pr(D \geq \beta)$ . Throughout the plateau region, stopping sets occur in the MP decoder and the ML decoder has linearly dependent columns in  $H_{\tilde{\mathcal{K}}}$  with probability very close to one. The results of Lemmas 4 and 5 give  $\Pr(D \geq 1) = 1 - \left(\frac{1-\epsilon}{1-\epsilon\delta}\right)^\eta$ , which can be examined in the limit as  $\eta \rightarrow \infty$ . It is immediate that except for when  $\delta = 1$  or  $\epsilon = 0$ ,  $\Pr(D \geq 1)$  goes to one for all  $(\delta, \epsilon)$  pairs as  $\eta$  gets large. From Theorem 1, if  $|R| = N - k$ , then



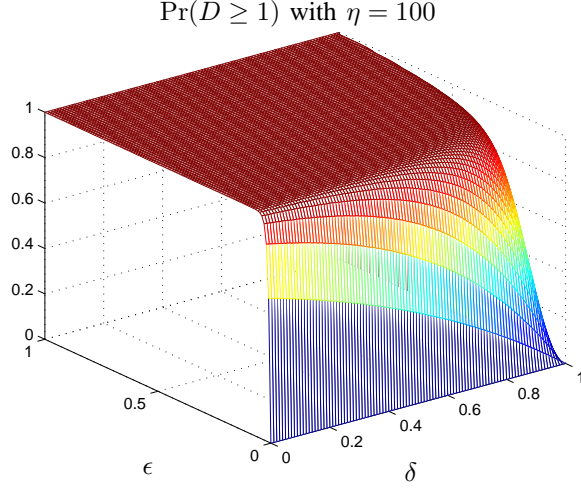


Fig. 5.  $\Pr(D \geq 1)$  when  $\eta = 100$ , as a function of the respective erasure probabilities in  $Q_m$  and  $Q_w$ ,  $\delta$  and  $\epsilon$ .

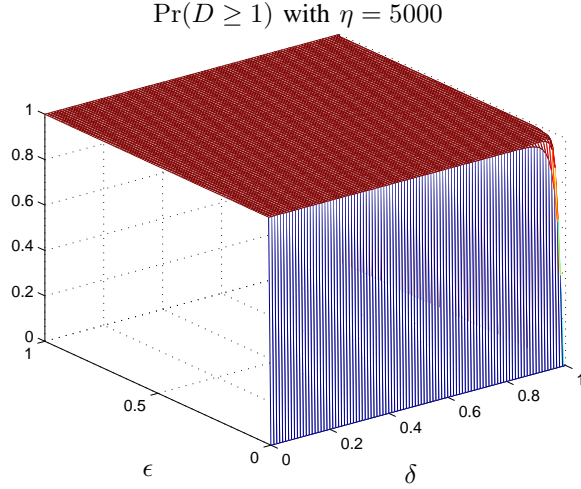


Fig. 6.  $\Pr(D \geq 1)$  when  $\eta = 5000$ , as a function of the respective erasure probabilities in  $Q_m$  and  $Q_w$ ,  $\delta$  and  $\epsilon$ .

$\eta = \frac{n}{\alpha} = \frac{k}{\alpha}$ . Clearly  $\eta$  grows with  $k$ ; therefore, the probability of security approaches one as  $k$  gets large. Since large  $k$  necessitates large  $n$  and  $N$ , the same holds true for these blocklength parameters. Codes with blocklength  $N = 10,000$  are deemed practical by today's standards. For  $\alpha = 1$  and for a carefully chosen  $R$  with size roughly 5000, then  $\eta \approx 5000$ . This case is shown in Fig. 6, where as expected, all nontrivial  $(\delta, \epsilon)$  pairs show  $\Pr(D \geq 1) \approx 1$ .

But of course, a single degree of freedom is easily guessed in an attack. Let us examine the effects on security when  $\beta$  takes on a larger value. This perspective is provided in Fig. 7, where  $\eta = 5000$  and  $\beta = 50$  with  $\alpha = 1$ . As can be seen in the figure, there exists a cutoff region, where  $(\delta, \epsilon)$  pairs within the plateau region will experience at least  $\beta$  degrees of freedom with probability very close to one, while pairs outside the region will have  $D < \beta$  with probability close to one. Owing to the severity of the cutoff, the threshold can be approximated

$\Pr(D \geq 50)$  with  $\alpha = 1$  and  $\eta = 5000$

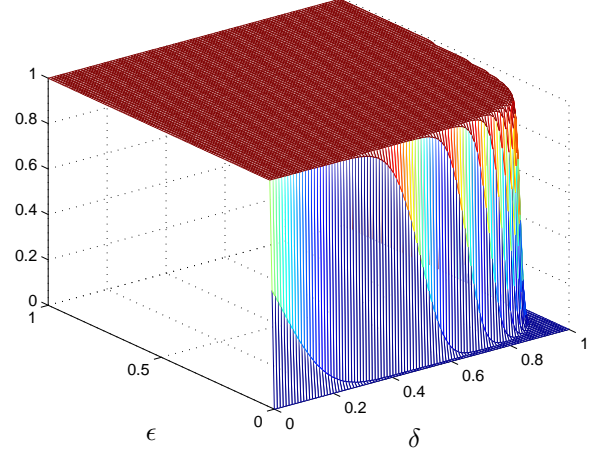


Fig. 7.  $\Pr(D \geq 50)$  when  $\alpha = 1$  and  $\eta = 5000$ , as a function of the respective erasure probabilities in  $Q_m$  and  $Q_w$ ,  $\delta$  and  $\epsilon$ .

by setting  $\Pr(D \geq \beta) = 0.5$  in (6), and deriving a function of  $\delta$  and  $\epsilon$ . This technique provides a unique threshold for each specific set of values for  $\beta$ ,  $\alpha$ , and  $\eta$ .

Finally, let us inspect the  $E[D]$  according to Theorem 1 for this case.

$$E[D] = \frac{\epsilon(1-\delta)}{1-\epsilon\delta} \eta \alpha = \frac{\epsilon(1-\delta)}{1-\epsilon\delta} n. \quad (11)$$

This function grows linearly with  $n$  which is equal to  $k$  when  $|R| = N - k$ . Thus, to drive  $D$  to a large number in practice, we simply must use a larger dimension in the encoder. Note that in the expectation the choice of  $\alpha$  does not affect security; although,  $\alpha = 1$  allows  $\eta$  to be as large as possible, which provides more confidence that  $D \approx E[D]$  by the law of large numbers ([41], pg. 193).

### C. Multiple Intended Receivers

In this section, we move past the single user case, and address the more general broadcast channel originally presented in [42]. There is also a single eavesdropper with probability of an erased packet equal to  $\epsilon$  as before. This case allows us to understand the repercussions on security of having more than one user for which we allow feedback requests. We can characterize security using Lemma 4 and Theorem 1 in the  $m$  user case by finding an expression for  $\Pr(R_{ef})$ . Recall that  $R_{ef}$  is the event that Eve receives a single transmitted packet as before. Let each user have an independent PEC with probability of erasure in the  $i$ th user's channel as  $\delta_i$  for  $i = 1, 2, \dots, m$ . The following lemma is necessary to obtain  $\Pr(R_{ef})$ .

**Lemma 6.** *If  $Q_1, Q_2, \dots, Q_m$  are independent geometrically distributed random variables with success parameters  $\lambda_1, \lambda_2, \dots, \lambda_m$ , and  $T_m = \max(Q_1, Q_2, \dots, Q_m)$ , then the probability mass function on  $T_m$  is given as*

$$f_m(t) = \prod_{i=1}^m (1 - (1 - \lambda_i)^t) - \prod_{j=1}^m (1 - (1 - \lambda_j)^{t-1}). \quad (12)$$

*Proof:* The proof is omitted for the sake of brevity, but follows from an inductive assumption on  $m$ . ■

Armed with this lemma, we can obtain  $\Pr(R_{ef})$  for the broadcast channel case.

**Lemma 7.** *Using the broadcast channel with  $m$  independent legitimate receivers and an eavesdropper*

$$\begin{aligned} \Pr(R_{ef}) &= \sum_{i=1}^m \left( \frac{1-\epsilon}{1-\epsilon\delta_i} \right) - \sum_{i<j} \left( \frac{1-\epsilon}{1-\epsilon\delta_i\delta_j} \right) + \\ &\quad \sum_{i<j<k} \left( \frac{1-\epsilon}{1-\epsilon\delta_i\delta_j\delta_k} \right) - \cdots + \\ &\quad (-1)^{m+1} \left( \frac{1-\epsilon}{1-\prod_{i=1}^m \delta_i} \right) \end{aligned}$$

where the notation  $i < j$  means the summation traverses over all pairs  $(i, j)$  such that  $i, j \in \{1, 2, \dots, m\}$  and  $i < j$ , and similarly for  $i < j < k$ , etc.

*Proof:* Note that if the  $i$ th user requests a single packet until it is received, and in each transmission it is received with probability  $\delta_i$ , then the total number of times the user must request the packet is governed by a geometric random variable with success parameter  $1 - \delta_i$  [41]. Define  $W_1, W_2, \dots, W_m$  as the geometric random variables governing the total number of transmissions necessary for users  $1, 2, \dots, m$ , respectively, to obtain the packet error-free. Then, let  $W = \max(W_1, W_2, \dots, W_m)$ .  $W$  governs the total number of transmissions necessary for all legitimate parties to receive the packet.

By Lemma 6, we know that

$$\Pr(W = w) = \prod_{i=1}^m (1 - \delta_i^w) - \prod_{j=1}^m (1 - \delta_j^{w-1}) \quad (13)$$

because the success parameter for  $W_i$  is  $1 - \delta_i$  for  $i = 1, 2, \dots, m$ . Finally, we point out that

$$\prod_{i=1}^m (1 - \delta_i) = 1 - \sum_{i=1}^m \delta_i + \sum_{i<j} \delta_i \delta_j - \sum_{i<j<k} \delta_i \delta_j \delta_k + \cdots + (-1)^m \prod_{i=1}^m \delta_i \quad (14)$$

which implies that

$$\begin{aligned} \Pr(W = w) &= \left( 1 - \sum_{i=1}^m \delta_i^w + \sum_{i<j} (\delta_i \delta_j)^w - \cdots + \right. \\ &\quad \left. (-1)^m \left( \prod_{i=1}^m \delta_i \right)^w \right) + \\ &\quad \left( -1 + \sum_{i=1}^m \delta_i^{w-1} - \sum_{i<j} (\delta_i \delta_j)^{w-1} + \cdots + \right. \\ &\quad \left. (-1)^{m+1} \left( \prod_{i=1}^m \delta_i \right)^{w-1} \right) \\ &= \sum_{i=1}^m \delta_i^{w-1} (1 - \delta_i) - \sum_{i<j} (\delta_i \delta_j)^{w-1} (1 - \delta_i \delta_j) \end{aligned}$$

$$+ \cdots + (-1)^{m+1} \left( \prod_{i=1}^m \delta_i \right)^{w-1} (1 - \prod_{i=1}^m \delta_i).$$

With these pieces in place, we commence proving the lemma.

$$\begin{aligned} \Pr(R_{ef}) &= \sum_{w=1}^{\infty} \Pr(R_{ef}|W = w) \Pr(W = w) \\ &= \sum_{w=1}^{\infty} (1 - \epsilon^w) \left( \prod_{i=1}^m (1 - \delta_i^w) - \prod_{j=1}^m (1 - \delta_j^{w-1}) \right) \\ &= \sum_{w=1}^{\infty} (1 - \epsilon^w) \left( \sum_{i=1}^m \delta_i^{w-1} (1 - \delta_i) - \sum_{i<j} (\delta_i \delta_j)^{w-1} (1 - \delta_i \delta_j) + \cdots \right. \\ &\quad \left. + (-1)^{m+1} \left( \prod_{i=1}^m \delta_i \right)^{w-1} (1 - \prod_{i=1}^m \delta_i) \right) \\ &= \sum_{i=1}^m \frac{1 - \delta_i}{\delta_i} \sum_{w=1}^{\infty} (1 - \epsilon^w) \delta_i^w - \\ &\quad \sum_{i<j} \frac{1 - \delta_i \delta_j}{\delta_i \delta_j} \sum_{w=1}^{\infty} (1 - \epsilon^w) (\delta_i \delta_j)^w + \cdots \\ &\quad + (-1)^{m+1} \frac{1 - \prod_{i=1}^m \delta_i}{\prod_{i=1}^m \delta_i} \sum_{w=1}^{\infty} (1 - \epsilon^w) \left( \prod_{i=1}^m \delta_i \right)^w \\ &= \sum_{i=1}^m \frac{1 - \delta_i}{\delta_i} \left( \sum_{w=0}^{\infty} \delta_i^w - \sum_{w=0}^{\infty} (\epsilon \delta_i)^w \right) - \\ &\quad \sum_{i<j} \frac{1 - \delta_i \delta_j}{\delta_i \delta_j} \left( \sum_{w=0}^{\infty} (\delta_i \delta_j)^w - \sum_{w=0}^{\infty} (\epsilon \delta_i \delta_j)^w \right) \\ &\quad + \cdots + (-1)^{m+1} \frac{1 - \prod_{i=1}^m \delta_i}{\prod_{i=1}^m \delta_i} \times \\ &\quad \left( \sum_{w=0}^{\infty} \left( \prod_{i=1}^m \delta_i \right)^w - \sum_{w=0}^{\infty} \left( \epsilon \prod_{i=1}^m \delta_i \right)^w \right) \\ &= \sum_{i=1}^m \left( \frac{1 - \epsilon}{1 - \epsilon \delta_i} \right) - \sum_{i<j} \left( \frac{1 - \epsilon}{1 - \epsilon \delta_i \delta_j} \right) + \cdots + \\ &\quad (-1)^{m+1} \left( \frac{1 - \epsilon}{1 - \prod_{i=1}^m \delta_i} \right). \quad (15) \end{aligned}$$

■

#### D. Collaborating Eavesdroppers

In this section we consider the case with  $l$  eavesdroppers working together in order to obtain the cryptogram  $M$ , each with a possibly unique probability of packet erasure  $\epsilon_1, \epsilon_2, \dots, \epsilon_l$ . All are assumed to obtain packets through independent PECs. It is simpler to first consider a single legitimate user Bob with probability of packet erasure  $\delta$ . Then the general result which assumes  $m$  friendly parties with  $l$  collaborating eavesdroppers comes easily.

**Lemma 8.** *For  $l$  eavesdroppers and a single legitimate re-*

ceiver,

$$\Pr(R_{ef}) = \frac{1 - \prod_{i=1}^l \epsilon_i}{1 - \delta \prod_{i=1}^l \epsilon_i}. \quad (16)$$

*Proof:* The proof is straightforward if we note that collaborating eavesdroppers receive a single sent packet if at least one of them obtains the packet error-free. Let  $W$  be a geometric random variable with success parameter  $1 - \delta$ . This governs the number of transmissions for each packet. Therefore,

$$\begin{aligned} \Pr(R_{ef}) &= \sum_{w=1}^{\infty} \Pr(R_{ef}|W=w) \Pr(W=w) \\ &= \sum_{w=1}^{\infty} (1 - (\prod_{i=1}^l \epsilon_i)^w) (1 - \delta) \delta^{w-1} \\ &= \frac{1 - \delta}{\delta} \left( \sum_{w=0}^{\infty} \delta^w - (\delta \prod_{i=1}^l \epsilon_i)^w \right) \\ &= \frac{1 - \prod_{i=1}^l \epsilon_i}{1 - \delta \prod_{i=1}^l \epsilon_i}. \end{aligned} \quad (17)$$

This answer provides an easy bridge to an extremely general result.

**Corollary 2.** *For the scenario with  $m$  intended parties and  $l$  eavesdroppers with similar notation as before,*

$$\begin{aligned} \Pr(R_{ef}) &= (1 - \epsilon') \left( \sum_{i=1}^m \frac{1}{1 - \epsilon' \delta_i} - \sum_{i < j} \frac{1}{1 - \epsilon' \delta_i \delta_j} + \cdots + \right. \\ &\quad \left. (-1)^{m+1} \frac{1}{1 - \epsilon' \prod_{i=1}^m \delta_i} \right), \end{aligned}$$

where  $\epsilon' = \prod_{i=1}^l \epsilon_i$ .

*Proof:* This proof is not included for the sake of brevity, but is nearly identical to the proof of Lemma 7 with slight alterations as indicated by the proof of Lemma 8 to allow for multiple eavesdroppers. ■

## VII. CRYPTOGRAPHIC SECURITY ENHANCEMENTS

The probabilistic security analysis in Section VI assumes that attacks on the cryptography become more difficult or completely infeasible as  $D$  gets large. It remains to show the effect of the coding scheme on attacks of the cryptography. As an example, fast correlation attacks on stream ciphers are known to be possible, even if the cryptogram is error-prone. It was noted in [8], [9], [10] that specific attacks from [7] were made more difficult, and in some cases impossible due to error rates in the cryptogram beyond a certain threshold. Certainly as bit error rates approach 0.5 in the cryptogram, attacks of the fast-correlation variety break down completely.

Let  $\hat{P} = (\hat{p}^1, \hat{p}^2, \dots, \hat{p}^L)$  be the collection of punctured codewords obtained by Eve, where  $\hat{p}^i = (\hat{p}_1^i, \hat{p}_2^i, \dots, \hat{p}_n^i)$ , and let  $\hat{B} = (\hat{b}^1, \hat{b}^2, \dots, \hat{b}^L)$  be the decoded codewords, where  $\hat{b}^i = (\hat{b}_1^i, \hat{b}_2^i, \dots, \hat{b}_n^i)$ . Finally, define the implied block structure of Eve's decoder output as  $\hat{M} = (\hat{m}^1, \hat{m}^2, \dots, \hat{m}^L)$ , where  $\hat{m}^i = (\hat{m}_1^i, \hat{m}_2^i, \dots, \hat{m}_k^i)$ . Each channel-erased bit in  $\hat{p}^i$

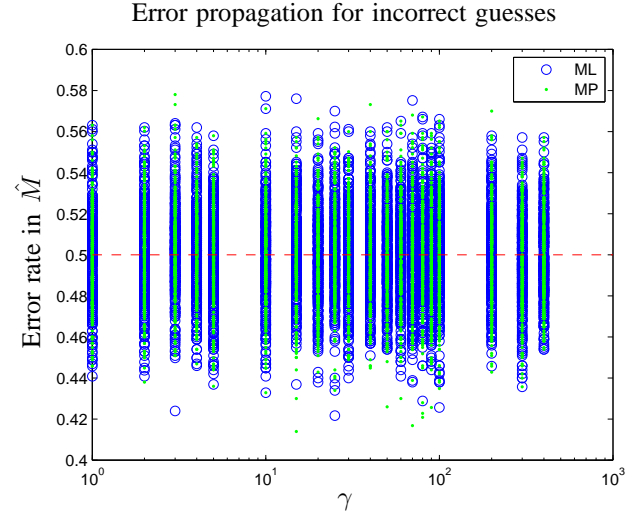


Fig. 8. The simulated error rates in Eve's decoded cryptogram  $\hat{M}$  when  $\gamma$  errors are made in guessing bit values for  $D$  degrees of freedom in Eve's received codewords.

yields a degree of freedom in  $\hat{b}^i$ , and complete recovery of  $\hat{b}^i$  requires that  $D$  bits in  $\hat{p}^i$  be guessed correctly. If a guess is incorrect, there will be at least as many errors in  $\hat{b}^i$  as the minimum distance of the LDPC code. The descrambling process in (4) magnifies any errors in  $\hat{b}^i$  to an expected bit error rate of 0.5 in  $\hat{m}^i$ . Therefore, since all guesses are equally likely, a brute-force attack on  $D$  bits must be accomplished to obtain each  $\hat{m}^i$ .

Simulations of the end-to-end encoder and decoder clearly (18) indicate the expected bit error rate in  $\hat{M}$  of 0.5 for an incorrect guess. Simulations were performed using the irregular LDPC code of Example 1 with  $N = 1000$  and  $k = 500$ . Puncturing patterns used were such that  $|R| \geq 498$  bits.  $S$  was formed randomly by setting roughly half of the  $k^2$  entries equal to one until such a matrix was invertible using the LU decomposition in GF(2). Let  $\gamma$  be the number of bits in Eve's guess which are incorrect. We offer simulation results for  $\gamma = 1, 2, 3, 4, 5, 10, 15, 20, 25, 30, 40, 50, 60, 70, 80, 90, 100, 200, 300$ , and 400 in Fig. 8. Each  $\gamma$  value was tested 300 times on both the MP and ML decoder, while a new puncturing pattern  $R$  was generated every 10 experiments, and a new code from the ensemble was selected every 30 experiments. All tests produced error rates in between 0.414 and 0.578 in  $\hat{M}$ , while the mean depicted a 0.5002 bit error rate with no noticeable difference between MP and ML decoders, or between  $\gamma$  values, as Fig. 8 indicates.

These results imply that unless  $D$  bits are guessed exactly, the cryptography must be attacked with an average bit error rate of 0.5 in  $\hat{M}$ . We can certainly expect such an attack to fail for fast correlation attacks on stream ciphers, but the notion that any attack on a cryptosystem could absorb such error rates and still succeed is obviously shortsighted. However, since an attack could feasibly be staged using a single block of  $\hat{M}$ , we will only guarantee failure of the attack if every block in  $\hat{M}$  is incorrect. Using similar logic, it can be said that if an attack would succeed using the error-free ciphertext  $M$ , then it may fail even if a single block in  $\hat{M}$  is in error.

**Theorem 2.** Define the complexity of a cryptographic attack to be  $C_A$ . Let  $D$  be the degrees of freedom of each of  $L$  blocks in  $\hat{B}$ . Then the expected complexity  $C_{PL}$  of a successful attack on the system is bounded as

$$2^{E[D]}(1 - 2^{-1/L})C_A \leq C_{PL} \leq 2^{E[D]}(2^{-1/L})C_A. \quad (19)$$

*Proof:* By Corollary 1 each codeword in  $\hat{B}$  has the same number of degrees of freedom. Thus,  $E[D]$  is the average number of bits that must be guessed in each of  $L$  punctured codewords in  $\hat{P}$ . Assume that an attacker guesses bit patterns on all codewords in  $\hat{P}$  simultaneously. The correct bit patterns of the channel-erased bits in the  $L$  codewords  $\hat{P}$  are uniformly distributed over  $2^{E[D]}$  possibilities in each block. The lower bound is formulated by the expected number of guesses until at least one of  $L$  codewords is found. Model the correct bit patterns in the  $L$  codewords as i.i.d. discrete uniform random variables on  $\{0, 1, \dots, 2^{E[D]} - 1\}$ , say  $U_1, U_2, \dots, U_L$ . Without loss of generality, assume that an attacker begins by guessing zero for each  $U_i$  and proceeds in an orderly fashion. Then, the expected number of guesses until at least one is correct is given by  $E[\min(U_1, U_2, \dots, U_L)]$ . Thus, we calculate  $\Pr(\min(U_1, U_2, \dots, U_L) \geq z) = \Pr(U_1 \geq z, \Pr(U_2 \geq z), \dots, U_L \geq z) = (\Pr(U_1 \geq z)(\Pr(U_2 \geq z) \dots (\Pr(U_L \geq z) =$

$$\left(\frac{2^{E[D]} - z}{2^{E[D]}}\right)^L. \quad (20)$$

Now, solve for  $z$  in  $\Pr(\min(U_1, U_2, \dots, U_L) \geq z) = 0.5$  for a close bound on the expectation to get the lower bound.

The upper bound is calculated similarly, but we assume that *all* patterns must be guessed in order to guarantee success, therefore, the bound is given by finding the  $z$  that solves  $\Pr(\max(U_1, U_2, \dots, U_L) < z) = 0.5$ . ■

As a check on these bounds, for  $L = 1$  we expect  $2^{E[D]-1}$  guesses on average for a successful attack. In this case, both bounds meet at  $2^{E[D]-1}C_A$ , as expected. Although these bounds are helpful, when  $L > 1$  the bounds are not as tight, and thus provide limited insight into the true increase in complexity of the attack. More than likely, an attack will require at least a certain number of consecutive blocks in  $M$  to execute successfully [7]. Clearly a 0.5 bit error rate in any block would destroy an attack with these requirements. Therefore, the upper bound in (19) serves as a good approximation to the expected amount of work necessary to complete the attack, with  $L$  being set by the attack specifications. Thus we see, that our system appends a multiplier which is exponential in  $E[D]$  to the complexity of a cryptographic attack through practical physical-layer security.

## VIII. CONCLUSIONS

In conclusion, we have presented the security metric of degrees of freedom  $D$  in an eavesdropper's received code-words, and applied this metric to a physical-layer coding scheme to show cryptographic security enhancements due to channel coding. The coding scheme relies on the nature of independent packet erasure channels and ARQ to provide secrecy and reliability, respectively. End-to-end details of the

encoder and decoder were provided. Design criteria were specified to maximize  $D$  in a maximum-likelihood attack as well as a message-passing attack. This involved security performance comparisons of LDPC codes with varying degree distributions, where irregular codes were shown to outperform regular codes in maximizing  $D$ . The expected value of  $D$  was also shown to be equal to  $H(X|Z)$  in our encoder. Probabilistic security results were obtained and made general so as to apply to multiple receivers and multiple collaborative attackers. Simulation results were provided which show that unless an attacker can guess  $D$  symbols in the received data correctly, the system yields a bit error rate of 0.5 in the cryptogram, thus necessitating a brute-force attack on  $D$  bits for each codeword. The end result on the expected increase in attack complexity on the cryptosystem due to our scheme is a multiplier which is exponential in  $E[D]$ . The system was shown to provide cryptographic security enhancement, even when eavesdroppers have an advantage over legitimate receivers in signal quality.

## REFERENCES

- [1] C. E. Shannon, "Communication theory of secrecy systems," *Bell Syst. Tech. J.*, vol. 28, no. 4, pp. 656–715, 1949.
- [2] A. D. Wyner, "The wire-tap channel," *Bell Syst. Tech. J.*, vol. 54, no. 8, pp. 1355–1387, Oct. 1975.
- [3] I. Csiszár and J. Körner, "Broadcast channels with confidential messages," *IEEE Trans. Inf. Theory*, vol. 24, no. 3, pp. 339–348, May 1978.
- [4] L. H. Ozarow and A. D. Wyner, "Wire-tap channel II," *Bell Syst. Tech. J.*, vol. 63, no. 10, pp. 2135–2157, Dec. 1984.
- [5] M. Bloch, J. Barros, M. R. D. Rodrigues, and S. W. McLaughlin, "Wireless information-theoretic security," *IEEE Trans. Inf. Theory*, vol. 54, no. 6, pp. 2515–2534, June 2008.
- [6] U. Maurer and S. Wolf, "Information-theoretic key agreement: From weak to strong secrecy for free," in *Advances in Cryptology — EURO-CRYPT 2000*, ser. Lecture Notes in Computer Science, B. Preneel, Ed., vol. 1807. Springer-Verlag, May 2000, pp. 351–368.
- [7] W. Meier and O. Staffelbach, "Fast correlation attacks on certain stream ciphers," *Journal of Cryptology*, vol. 1, pp. 159–176, 1989.
- [8] W. K. Harrison and S. W. McLaughlin, "Physical-layer security: Combining error control coding and cryptography," in *Proc. IEEE Int. Conf. Communications (ICC)*, Dresden, Germany, June 2009, pp. 1–5.
- [9] —, "Tandem coding and cryptography on wiretap channels: EXIT chart analysis," in *Proc. IEEE Int. Symp. Information Theory (ISIT)*, Seoul, Korea, June-July 2009, pp. 1939–1943.
- [10] —, "EXIT charts applied to tandem coding and cryptography in a wiretap scenario," in *Proc. IEEE Information Theory Workshop*, Taormina, Sicily, Oct. 2009, pp. 173–177.
- [11] W. K. Harrison, J. Almeida, D. Kline, S. W. McLaughlin, and J. Barros, "Stopping sets for physical-layer security," in *Proc. IEEE Information Theory Workshop (ITW)*, Dublin, Ireland, Aug.-Sept. 2010, pp. 1–5.
- [12] M. Baldi, M. Bianchi, and F. Chiaraluce, "Non-systematic codes for physical-layer security," in *Proc. IEEE Information Theory Workshop (ITW)*, Dublin, Ireland, Aug.-Sept. 2010, pp. 1–5.
- [13] D. Kline, J. Ha, S. McLaughlin, J. Barros, and B.-J. Kwak, "LDPC codes for the Gaussian wiretap channel," in *Proc. IEEE Information Theory Workshop (ITW)*, Taormina, Sicily, Oct. 2009, pp. 95–99.
- [14] D. Kline, J. Ha, S. W. McLaughlin, J. Barros, and B.-J. Kwak, "LDPC codes for physical layer security," in *Proc. IEEE Global Telecommunications Conf. (GLOBECOM)*, Honolulu, HI, Nov. 2009.
- [15] M. Bloch, R. Narasimha, and S. W. McLaughlin, "Network security for client-server architecture using wiretap codes," *IEEE Trans. Inf. Forensics Security*, vol. 3, no. 3, pp. 404–413, Sept. 2008.
- [16] Y. Liang, H. V. Poor, and L. Ying, "Secrecy throughput of MANETs with malicious nodes," in *Proc. IEEE Int. Symp. Information Theory (ISIT)*, Seoul, Korea, June-July 2009, pp. 1189–1193.
- [17] A. Thangaraj, S. Dihidar, A. R. Calderbank, S. W. McLaughlin, and J.-M. Merolla, "Applications of LDPC codes to the wiretap channel," *IEEE Trans. Inf. Theory*, vol. 53, no. 8, pp. 2933–2945, Aug. 2007.

- [18] L. Lai, H. El Gamal, and H. Poor, "The wiretap channel with feedback: Encryption over the channel," *IEEE Trans. Inf. Theory*, vol. 54, no. 11, pp. 5059–5067, Nov. 2008.
- [19] M. A. Latif, A. Sultan, and H. El Gamal, "ARQ-based secret key sharing," in *Proc. IEEE Int. Conf. Communications (ICC)*, June 2009, pp. 1–6.
- [20] A. T. Suresh, A. Subramanian, A. Thangaraj, M. Bloch, and S. W. McLaughlin, "Strong secrecy for erasure wiretap channels," in *Proc. IEEE Information Theory Workshop (ITW)*, Dublin, Ireland, Aug.-Sept. 2010, pp. 1–5.
- [21] E. Arkan, "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," *IEEE Trans. Inf. Theory*, vol. 55, no. 7, pp. 3051–3073, July 2009.
- [22] E. Hof and S. Shamai, "Secrecy-achieving polar-coding for binary-input memoryless symmetric wire-tap channels," *Submitted to IEEE Trans. Inf. Theory*, Available online at [http://arxiv.org/PS\\_cache/arxiv/pdf/1005/1005.2759v2.pdf](http://arxiv.org/PS_cache/arxiv/pdf/1005/1005.2759v2.pdf), Aug. 2010.
- [23] H. Mahdaviifar and A. Vardy, "Achieving the secrecy capacity of wiretap channels using polar codes," *Submitted to IEEE Trans. Inf. Theory*, Available online at [http://arxiv.org/PS\\_cache/arxiv/pdf/1007/1007.3568v1.pdf](http://arxiv.org/PS_cache/arxiv/pdf/1007/1007.3568v1.pdf), July 2010.
- [24] M. Bloch and J. Barros, *Physical-Layer Security: From Information Theory to Security Engineering*. To Appear: Cambridge University Press, 2010.
- [25] D. R. Stinson, *Cryptography Theory and Practice*, 3rd ed., ser. Discrete Mathematics and Its Applications, K. H. Rosen, Ed. Boca Raton, FL: Chapman & Hall/CRC Taylor & Francis Group, 2006.
- [26] U. Maurer and S. Wolf, "Secret-key agreement over unauthenticated public channels—Part I: Definitions and a completeness result," *IEEE Trans. Inf. Theory*, vol. 49, no. 4, pp. 822–831, Apr. 2003.
- [27] R. G. Gallager, *Low-Density Parity-Check Codes*. Cambridge, MA: MIT Press, 1963.
- [28] T. K. Moon, *Error Correction Coding: Mathematical Methods and Algorithms*. Hoboken, NJ: John Wiley & Sons, Inc., 2005.
- [29] T. Richardson and R. Urbanke, *Modern Coding Theory*. New York, NY: Cambridge University Press, 2008.
- [30] D. Burshtein and G. Miller, "An efficient maximum-likelihood decoding of LDPC codes over the binary erasure channel," *IEEE Trans. Inf. Theory*, vol. 50, no. 11, pp. 2837–2844, Nov. 2004.
- [31] T. J. Richardson and R. L. Urbanke, "Efficient encoding of low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 638–656, Feb. 2001.
- [32] T. K. Moon and W. C. Stirling, *Mathematical Methods and Algorithms for Signal Processing*. Upper Saddle River, NJ 07458: Prentice-Hall, Inc., 2000.
- [33] K.-M. Lee and H. Radha, "The design of the maximum-likelihood decoding algorithm of LDPC codes over BEC," in *Proc. 41st Annu. Conf. Information Sciences and Systems*, Baltimore, MD, Mar. 2007.
- [34] C. Di, D. Proietti, I. E. Telatar, T. J. Richardson, and R. L. Urbanke, "Finite-length analysis of low-density parity-check codes on the binary erasure channel," *IEEE Trans. Inf. Theory*, vol. 48, no. 6, pp. 1570–1579, June 2002.
- [35] E. Rosnes and O. Ytrehus, "An efficient algorithm to find all small-size stopping sets of low-density parity-check matrices," *IEEE Trans. Inf. Theory*, vol. 55, no. 9, pp. 4167–4178, Sept. 2009.
- [36] A. Orlitsky, K. Viswanathan, and J. Zhang, "Stopping set distribution of LDPC code ensembles," *IEEE Trans. Inf. Theory*, vol. 51, no. 3, pp. 929–953, Mar. 2005.
- [37] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Hoboken, NJ: John Wiley & Sons, Inc., 2006.
- [38] A. Alloum, J. J. Boutros, G. I. Shamir, and L. Wang, "Non-systematic LDPC codes via scrambling and splitting," in *Proc. Allerton Conf.*, Monticello, IL, Sept. 2005, pp. 1879–1888.
- [39] G. I. Shamir and J. J. Boutros, "Non-systematic low-density parity-check codes for nonuniform sources," Adelaide, South Australia, Sept. 2005, pp. 1898–1902.
- [40] A. Konheim, "A queueing analysis of two ARQ protocols," *IEEE Trans. Commun.*, vol. 28, no. 7, pp. 1004–1014, July 1980.
- [41] G. Grimmett and D. Stirzaker, *Probability and Random Processes*, 3rd ed. Oxford, UK: Oxford University Press, 2001.
- [42] T. Cover, "Broadcast channels," *IEEE Trans. Inf. Theory*, vol. 18, no. 1, pp. 2–14, Jan. 1972.